ASEPH
Appendix G

Glossary

Version 1.0

February 28, 2000

activity - a series of purposeful actions performed to achieve specific objectives of a process. For example, designing a module and reviewing a report. Activities consume and produce artifacts.

action item - an action item describes some task associated with a project to allow the task to be assigned and tracked. Tasks can be pre-planned or unplanned (unexpected). Planned tasks are major milestones and scheduled work packages. Unexpected tasks analyze and solve problems. An action item may refer to any unresolved issue or pending task of interest to the project. Generally, action items are exceptional events that can jeopardize project success. Action items are tracked until they are completed.

action item list - a list of actions that need to be performed on a project. The list should show the action, who is responsible for the action, the date the action is due to be completed and a level of priority for the action.

allocated baseline - approved documentation describing an item's functional, interoperability and interface characteristics that are allocated form the approved system level configuration.

analysis - the activity of determining the requirements for a product. To create a totally new product, a specification is either provided by the buyer or is written by the developer (sometimes jointly with customer representatives). To modify an existing product, changes to the existing (baseline) product are defined by analyzing Baseline Change Requests (BCRs) and Software Trouble Reports (STRs). [STRs are sometimes called Product Trouble Reports or PTRs.] In cases where the baseline requirements are not recorded, they are reconstructed by reverse engineering the source code and/or by analyzing existing documentation. For large projects, this activity ends with the System Requirements Review (SRR) or the Software Requirements Review (SSR).

annual training plan - this plan is prepared annually and defines the organization's planned schedule for the course development and delivery during the coming year.

approved vendor list - a list of vendor that have met certain criteria for doing business with SAIC and are currently in good standing with SAIC.

archetype - a standard model/abstraction representing the essential feature of a particular type of project. For example, DV represents SDV, HDV and FCD.

architecture - see software architecture

artifact - a tangible object produced by an activity. Examples are specifications, design documents, audit records, technical reports, plans, schedules, and training courses.

ASEPH compliance guide – Shows where each item and artifact comprising the project record is located. The ASEPH Compliance Guide will also be used to record the tailoring of the standard process (e.g., by identifying customer-specified documents that replace ASEPH standard documents). The ASEPH Compliance Guide is used by Quality Assurance auditors to locate required process artifacts.

audit summary report - a report written by Quality Assurance summarizing the results of the audits performed on each project.

baseline - a formally approved version of a configuration item or items, regardless of media, formally designated and fixed at a specific time during the configuration items' life cycle.

A specification, document, software element, hardware item or product that has been formally reviewed and/or agreed upon; it thereafter serves as the basis for further development and can only be changed through formal change control procedures.

The foundation for configuration management. It provides the official standard on which subsequent work is based and to which only authorized changes are made. After an initial baseline is established and frozen, every subsequent change is recorded as a delta until the next baseline is established.

baseline change request (BCR) - a request submitted by an end user or Project Office for a new or modified capability. The BCR defines additional requirements and/or modifies existing (baseline) requirements. BCRs is a general name. Often such requests arrive as Change Requests, Trouble Reports, etc. Since BCRs arrive in many forms, all are recorded at the SED as Engineering Change Requests (ECRs) for tracking purposes.

bill of materials(BOM) - a list of the materials needed for a project containing a description, quantity and price.

build - a functional software product, usually delivered to the end user. Also called a release or a version. We often use the noun "build" to refer to an increment or a release. Some authors also distinguish the terms "version" and "release". Version refers to a product with major enhancements. Release refers to products with minor changes, especially corrections to defects found in the preceding version. So, each version is followed by one or more releases.

build implementation plan (BIP) - a form of PIP for software products. It identifies the specific software components to be built (using the tailored process defined in the PSP), the milestones, the schedule and the resources required. Key parts of the Build Implementation Plan are:

(1) The list of deliverable items
(2) The Project Schedule (milestones and planned completion dates).
(3) Specific risks identified for the build
(4) Special training needed by the project staff
(5) The estimated resources (labor hours, Bill of Materials identifying items to be purchased).

(6) Modifications to the system's SDP made for this specific build/version.
(7) Responsible person(s)

A more general name for the BIP is the Product Implementation Plan (PIP). (The term "Build" is software-specific.)

build directive - a document that identifies the components and scripts used to produce a build (see S-CM-070).

build types - the build types are defined in this glossary they are:

- Normal Build
- Engineering Release Build
- Emergency Build
- Prototype Build
- Demonstration Build

business process - the sequence of activities "enclosing" the production process. These activities are common to all types of products and process, and include defining the job, negotiation with the customer and reporting project status.

buyer - the customer person or department responsible for the contractual issues in buying a product. This is usually the Project Office for systems built by the Government.

capability maturity model (CMM) - a conceptual framework based on state-of-the art software engineering practices to help organizations:

- characterize the maturity of their process
- establish goals for process improvement
- set priorities for immediate actions
- envision a culture of software engineering excellence

Five levels of process maturity are defined with each having specific Key Process Areas. Common themes of the CMM are: defined, documented, trained, practiced, measured, maintained, continuously improved, supported, and enforced. The application of the CMM is achieved by: appraisals (assessments, evaluations) and process improvement.

ccb minutes - the minutes from a meeting of the Configuration Control Board. These minutes should identify the MRs that where discussed and the decision of the board concerning each one, as well as a list of attendees.

change notification - a notice (either electronic or paper) informing affected personnel that a document has been altered so they can obtain an updated copy.

change request - any request submitted by a customer (PMO or users) for a change to alter the system. These appear in various forms and include Software Trouble Reports (STRs) and Baseline Change Requests (BCRs). BCRs request changes to the system's specification.

code and unit test (CUT) - the activity of generating working code (manually or using various tools) to record the data structures, algorithms and control logic defined during Detailed Design. The source code is compiled, linked and executed in small units to verify their correct operation. Also see SWIT. (Test procedures and tools for SWIT and FAT are typically developed concurrently with Code and Unit Test.)

configuration management strategy - defines the nature and scope of the CM activities needed for the project. It is prepared in ENG1 and used to estimate the resources needed. It is also used in ENG2 to write the actual plans.

configure - the activity of characterizing a project, selecting and adapting the standard process, and documenting the resulting process in the Project Management Plan.

configuration item - an entity (hardware component, subsystem, CSCI, document, data items, etc.) within a configuration that satisfies an end use function and that can be uniquely identified at a given reference point.

Items that are placed under configuration management and treated as a single entity.

contract compliance sheet - is a table testing all key contract specifications (technical, cost and schedule) to help control a proposal and the subsequent project, and to identify areas of risk. It is required by Policy A-20 for all Firm Fixed Price contracts. (It is recommended for any large contract.) It is usually created by extracting "shalls", "wills", and "musts" from the customer's RFP package. See SAIC Policy A-20 for the format of the Contract Compliance Sheet.

correspondence file - a file of all written communications between the project and the Customer maintained by each project. This is important to maintain continuity if personnel leave the project, and may also serve to support engineering change proposals and claims for reimbursement. The Contracts Department maintains the official file of all contractual documents and letters. The Project Manager is responsible for establishing a supplemental file containing documents such as letters, memos, etc. These documents are associated with the projects operation and thus are not part of the usual product documentation. The Project Manager is responsible for defining a numbering scheme and procedure for distributing and filing these documents. On small projects this can be handled with SAIC Administrative procedures. On larger projects, separate procedures may need to be written.

Much communication occurs on a Contract via telephone, FAX and email. Most of this information is transient in nature and there is no need to log and file most of it. The Project Team should use their best judgment to decide which items should be filed with additional guidance from the Contracts and Legal departments. Most phone calls should not be logged or documented. In certain cases, however, the issues discussed could be important enough to warrant documenting the conversation in the form of a memo to communicate the information to the rest of the staff and provide a historical record.

When users report problems with fielded products, the problems should be recorded on Modification Requests (MRs) forms or should be logged in the ATISD problem reporting system operated by the Customer Support help desk. (The Customer

Support concept does not exist yet but is being defined. If it is implemented, appropriate procedures and standards will be defined.)

COTS decision - a recording of the decisions made by a project team concerning the use of COTS products on their project. The document should state what products were considered, what products were accepted, the criteria used to select the products, and other relevant rationale and data.

course critique - a form used to express the opinion of the students about the effectiveness of course material and the instructor. The information is collected for a class and sent to the Corporate Training personnel.

course description - a brief description of an SAIC course which states the basic material covered by the course, the prerequisites for the course and who is required to take the course.

course material - the student training guide and any other additional materials such as articles, exercises, video tapes or visual aids needed to teach a course.

course need - the request from any employee for a course they fill needs to be created or modified. This request is used as input to the course creation/modification process.

course/role matrix – a matrix that shows the courses that are required for each and every role defined in the ASEPH. It is based on the Common Approach matrix but has been extended to cover the ASEPH roles.

corporate training program catalog – a catalog published by the corporation which contains a description of the course and the material that it covers.

critical design review - a review that takes place at the end of detailed design before actual production or coding begins.

customer - either the Buyer or End User of a product.

customer supplied material list - a list of materials needed for a project that the customer will provide. The project is not responsible for obtaining the items the Customer will obtain them for the project**.**

defect - a manifestation of a (possible) fault. We distinguish "raw" trouble reports and validated, unique trouble reports. This activity is performed since some trouble reports are due to equipment failures, misuse, etc. Also, some problems may be reported by multiple user sites. The screening activity "consolidates" the "raw" trouble reports to extract the ones relating to the software and its documentation.

delivery - products and services to be delivered are well-defined and produced (build and buy done).

demonstration build - a Demonstration Build is a software build that may or may not eventually be used as part of a fielded system. Demonstration Builds are designed to show the particular capability of a product or an organization. Product capability demonstrations should be treated as a Prototype Build. Organizational capability demonstrations may be a part of a Transition project. Often, this Demonstration is called a Capability Demonstration Build. A Capability Demonstration Build is a software build that reproduces the last Prime Item Developer software product to show the customer that ATISD has the capability to sustain the software product. The process for Transition projects, should be implemented prior to producing the Capability Demonstration Build. If a Capability Demonstration Build is to be used as part of a fielded system, it is considered a Maintenance project and the process for Maintenance projects, should be implemented.

deployment - we package items, ship, inspect at site, install and checkout, have Site Acceptance Test, convert databases, train users, and perform "logistics" tasks [provide spare parts, provide depot maintenance capability, negotiate maintenance agreements]. These tasks are typically off premises, and involve one or (usually) multiple sites. We are "putting it into service" or activating a unique or very complex system - something the user lacks the skills/knowledge to do.

deployment plan - a plan to deliver and install a new product version at the users' sites. The degree of SAIC participation in this activity varies. In some cases SAIC merely delivers a Technical Data Package. In other, staff actually visits each site, installing the software and training the users. Sometimes called a Fielding Plan.

detailed design (DD) - the activity of defining the data structures, algorithms and control logic which meet the functional and performance requirements allocated to a component or module and which comply with the constraints of the software architecture. For large projects, this activity ends with the Critical Design Review (CDR).

detailed estimates – estimates of software size, task effort, materials, ODCs, schedule, and product quality. These are prepared by the Project Team during ENG1 and are used to prepare the Offer in PM3.

developer - the organization that is building the product, in the content of the ASEPH this is typically SAIC.

directives - may be meeting minutes, redlined briefing charts, contract letter, change pages to specifications, contract modifications, etc.

distribution process - a description of how a product will be sent out to the various sites that SAIC is required to service. It should list the versions of the product, the sites to receive the products who is responsible for the various activities of distribution, and a schedule for distribution.

duration - time expended to perform some activity. Measured in work-days (also called project-days). Typically, engineers record calendar dates and the data analysis staff compute the work-days.

effort - labor expended to perform some activity. Measured in person-hours.

emergency build - the customer indicates an Emergency by demanding a software build with typically a few specific requirements and a very short time period for completion. Also, the delivery date specified cannot be slipped. It is a "The product must be in the field no later than a week from today" situation. No negotiation is possible. Reasons for an Emergency Build may be a war or conflict between U.S. and foreign powers or a soldier safety issue.

During an Emergency Build, the project may need to tailor out and/or defer some usual or normal process activities. The Analysis, Design, Build and/or Test process activities may be drastically shortened due to the project circumstances. An Emergency Build is done quickly, without thorough analysis. The Project Team however, must reach consensus on project tailoring and execution.

The next Normal Build must include complete analysis, documentation and extensive testing of the Emergency modifications. The additional work is done to ensure that the integrity of the system's design is not eroded by inadequately analyzed or evaluated modifications. The cost and risk of an Emergency Build is thus higher due to this reanalysis and rework. A positive aspect of this high cost is, in a sense, that it helps ensure that customers do not declare every build to be an Emergency.

end user - the person or department in the customer organization who will use a product.

engineering change analysis (ECA) - an SED form to record the results of analysis of an ECR.

engineering change request (ECR) - an SED form to record receipt of a customer request.

engineering change order (ECO) - an SED form documenting the set of ECR/ECAs to be included in a build. The ECO defines the build content.

engineering process - same as production process

engineering release build - the Engineering Release Build is a release of the software build to the customer prior to formal approval of the Engineering Change Proposal (ECP) or Engineering Release Record (ERR). The build is internally baselined by Configuration Management prior to release to the customer.

equipment list - a list of all the equipment used by a project that must be calibrated. The list should contain the following information about each piece of equipment: Equipment Name, Identifying Number (may be serial number, etc.), Location (e.g., off site equipment, at receiving inspection, etc.), Calibration Interval (this is established by the Chief Engineer based upon the manufacturer's recommendation, usage, criticality, etc.), Calibration Date, Due Date.

estimate - a figure in money, time or size that is the "best guess" at what something will cost, how long it will take to complete or how large or small it will be. This "best

guess" should be based on something such as historical data or an approved estimating technique.

evolutionary - a way to describe the development of a product. The complete design or requirements are not know when the project begins but it "evolves" as the project progresses.

field test (FT) - a test conducted outside the developer's facility to verify correct operation under field conditions. Usually done to demonstrate capabilities. Also see Site Acceptance Test.

fielding strategy - a description of how a product will be "put" in the "field". It should address the activities that are required, who is responsible for each activity and a schedule.

final test report - a report prepared by Quality Assurance at the end of FAT or SAT detailing the results of the acceptance test.

first article - the first item completed and accepted by the customer. Typically refers to hardware. Production of duplicate articles can begin once the first article has been accepted.

formal acceptance test (FAT) - a test conducted in the developer's facility to verify that the product meets all its acceptance criteria. "Formal" means that a documented procedure is executed in the presence of customer witnesses. Also called Formal Qualification Test (FQT) or Factory Acceptance Test (FAT).

functional baseline - approved documentation describing a system's function, interoperability and interface characteristics and the verification required to demonstrate the achievement of those specified characteristics.

gantt chart - a chart which represents project activities as horizontal lines on a calendar oriented chart, with triangles to denote major milestones associated with the activity. Progress is indicated by filling in the triangles when milestones are completed. Gantt charts are useful for simple schedules, but do not show task dependencies. Resource loaded networks (RLNs) should be used for projects with many interdependent tasks.

hard deliverables - products and services whose requirements are defined by the customer. Usually, these requirements are very specific and impose demanding constraints which are difficult to satisfy.

increment (build) - a working version of the product with a SUBSET of the system's total required capabilities. It is built as a milestone to show tangible progress (e.g., that certain risks or problems have been solved). Engineering builds are often demonstrated to the buyer to show progress (and maybe get feedback.) It will become part of the final product by (1) adding new pieces, (2) extending/evolving the current pieces, or (3) both.

incremental - the product is developed in stages or increments. These increments may be delivered to the Customer or they may be treated as demonstrations of developed capability.

implementation - Detailed Design plus Code and Unit Test. These are usually treated as a unit since a single programmer does these activities for a particular module or component. Implementation is usually very short for sustainment projects since the amount of code being written or modified is small.

initial estimates – estimates of the scope of a project prepared by the Project Manager during PM2. These estimates are used to determine the resources for ENG1, and as a starting point for the Detailed Estimates.

installation - the phase of a Transition project which executes the Transition Plan to establish the necessary facilities, define the sustainment process and place the baseline products under configuration control.

ISO - ISO 9000 is a set of five universal standards for a Quality Assurance system that is accepted around the world. Currently 90 countries have adopted ISO 9000 as national standards. When you purchase a product or service from a company that is registered to the appropriate ISO 9000 standard, you have important assurances that the quality of what you receive will be as you expect.

The most comprehensive of the standards is ISO 9001. It applies to industries involved in the design and development, manufacturing, installation and servicing of products or services. The standards apply uniformly to companies in any industry and of any size. Many companies require their suppliers to become registered to ISO 9001 and because of this, registered companies find that their market opportunities have increased. In addition, a company's compliance with ISO 9001 insures that it has a sound Quality Assurance system, and that's good business.

Registered companies have had dramatic reductions in customer complaints, significant reductions in operating costs and increased demand for their products and services.

ISO 9000 registration is rapidly becoming a must for any company that does business in Europe. Many industrial companies require registration by their own suppliers. There is a growing trend toward universal acceptance of ISO 9000 as an international standard.

item complete - this is a milestone used for hardware. It indicates that the "Assembly (fabricated item) conforms to its Assembly Drawing". That is, the item is completely fabricated and assembled. It is now ready for subsequent verification and integration activities such as (final) inspection (form, fit, function, finish), power up testing (smoke test), integration with other assemblies and then testing of the (sub) system.

job assignment - maps role to person

job descriptions - textural description of the available job opening. Should specify job title, qualification needed, length of job and category.

labor estimates - states the labor hours required to accomplish a project. Should break the hours down into engineering, Quality Assurance, Configuration Management, Peer Reviews, Project Management and any other activities required.

labor grade breakdown - breaks down the hours of an estimate into "grades" or categories often based on Customer requirements such as experience or education. The grades have different pay scales an so affect the cost estimate.

list of deliverables - a list of the items (documents, hardware, software, etc.) that are to be given to the Customer as part of the project.

list of openings - a list generated by the Project Manager describing job openings for their project. This list will be used by Human resources to post the job openings and search for suitable job candidates.

list of risks - a temporary informal list of potential risk that are gathered during different phases of the project. These risks will latter be consolidated and "formally" recorded on the Risk Worksheet.

meeting minutes - a description of a meeting that has occurred. Should contain a list of attendees, when the meeting was held, where the meeting was held, the subject of the meeting, any important things discussed and any action items assigned or closed.

normal build - a Normal Build is a software build with specified requirements, schedule and cost, which have been negotiated and agreed upon by SAIC and the customer. The requirements, schedule, cost, etc., are more flexible than for an Emergency Build. Typically, the build content (modifications to be made) is adjusted to fit the specified and/or some specified cost.

During a normal build, SAIC follows all normal process activities and procedures consistent with the ATISD policy, process, procedures, and standards. Individual project process tailoring is considered to be normal business and follows the tailoring procedures.

offer - a short plan summarizing the information contained in the project plan(s). This information includes the deliverables, schedule and estimated resources. (Often, the Offer is prepared and presented to the Customer before the actual Project plan(s) is written. The information is produced by the "Define the Approach" activity in the standard Project Management process.) Used to negotiate the scope and funding for a project.

operational concept - the Operational Concept describes the environment in which the product will operate and how the users will operate it after it has been installed. It should represent the consensus between the users, buyers, developers, and maintainers.

The Operational Concept describes the mission (purpose) of the system and its operational and support environments. Is also describes the functions and characteristics of the system within the overall system. It is written from a user standpoint and is non-technical.

The Operational Concept is usually prepared by the System Engineers. It is prepared after defining the system's mission and the top level system requirements. (In fact, these two items are usually included as part of the Operational Concept itself.) It is prepared prior to writing the specifications for subsystems, hardware components, software components, etc.

A system consists of hardware, software, data bases, policies, procedures, etc. plus the skilled humans who operate it. A system integrates two domains: product and operator. The key function of the Operational Concept is to clearly delineate which domain will perform the functions needed to perform the system's mission. (This is just the first step in requirements decomposition and allocation for the project.) The Operational Concept identifies those system functions (or decomposition thereof - the inevitable intertwining of specification and design) which are performed manually (by the operators) and which are performed automatically (by the product). This allocation is made to system components (implemented with hardware, software and data) and roles (performed by skilled humans who have been programmed with specific training).

This allocation of system requirements is important because it determines the size and complexity of the product, the skill level of the operators, and the amount of training that must be provided. This allocation has a crucial effect on the cost to develop, test and operate the system. (For example, automating more functions increases development costs but decreases crew size, thereby reducing labor costs during operation.)

The Operational Concept (Ops Concept or OPSCON) is also called the Concept of Operation (CONOPS). Both terms are used interchangeably. Note, however, that the term "Concept of Operations" is also used to denote the technical principles of operation of a technical device. Hence we prefer the term Operational Concept.

organizational chart - a chart showing how the organizational structure is set up for a project. It should show from the Group Manager to Administrative Support personnel.

organization status report – a report that indicates the status of the training program (courses required, courses taken, etc.) for any given organizational lever (Group, Operation, Division)

original estimates - estimates developed by the engineers and the Project Manager during ENG1 “Define the Approach”.

packing list - a document listing the items included in a shipment. It should contain a description, quantity and if appropriate a unique identifier for each item.

phased - the product is developed in phases or parts. Typically the product is divided along functional lines and the parts are delivered and used by the Customer.

phone log - a record of discussions between the Project Team and the Customer. Should specify who was involved in the conversion, the topic discussed and when the conversation took place.

plan - a plan defines a set of activities and tasks to accomplish some desired end, such as making a product. A plan describes what is to be done, by whom and how the Project Manager intends to organize the work, deploy the resources and control the performance of the tasks. The purpose of a plan is to record this information so it can be communicated to all participants who then use the information as a guide for their own particular decisions and actions.

Plans can have different levels of detail. For engineering projects, a plan is primarily a high level document, used to coordinate the work of several organizations and performers. The detailed activities are described by procedures which are referenced by the plan. The plan thus identifies a set of procedures and their sequence of execution. Because the procedures are separated it becomes possible to "reuse" them on multiple projects. This is a key tenet underlying efforts to define standard processes for an organization.

A new dimension arises when an organization (i.e., a project team) produces successive versions of a given product using the same production process. There are parts of a plan that are specific to a particular version and other parts which are common to all versions of the product (e.g., a weapon system or product line). In the past (before the idea of standard process come into vogue), projects were defined in an ad hoc way. Plans were built from scratch for each project and were not reusable. Thus, the process definition (procedures, standards and process flows) and the Project Management Plan (deliverables, schedule, resources) were merged in a single document. (Sometimes even the product description were included.) It make sense to structure or plans so that we can easily reuse the large amounts of repeated information.

preliminary design review - a review of the top level design of the product.

procedure - a written sequence of steps to guide an individual in performing some activity.

process - the collections of activities, methods, techniques and tools used to define and produce some product, including interim products.

process architecture - the sequencing of the production activities (analyze, design, code, test) chosen to produce a product or a series of related products. Examples are the waterfall, incremental, phased, evolutionary and spiral development models. (These are called project life cycles.)

process documentation - serves three purposes. First, it records the engineering information to make it visible and to exchange information between members of the design team. Second, it captures information needed by the maintainers to modify the system later. Third, it makes progress visible so that the customer and management can verify that progress is being made. All three types of documentation will decrease once case tools with a central repository of design information come into use. Thus, documentation costs can be expected to decrease.

process flow - a process flow is a diagram depicting the sequence of activities performed for a specific project type. The activities are represented by rectangles

("boxes"). The boxes are placed in columns to indicate who performs the activity depicted by the box. The boxes are linked by lines to indicate the sequence in which the boxes are executed (performed). The lines also represent the flow of data between the activities. Specific products are associated with each line. For example, a Project Manager may write a list of tasks which are then given to the engineers to perform.

process selection - choosing and (possibly) tailoring the process to be used to build (or sustain) a product. The process includes methods, techniques, tools, and staff skills. Often, additional intermediate products are identified. These products may be deliverable (e.g.,, progress reports) or non-deliverable (e.g., files of dummy data used for testing).

product - is a well defined, tangible object such as software, documents, databases, hardware items, etc. may be delivered to the customer or can be generated and used in the production process.

product design (PD) - the activity means different things for New Development and for Maintenance. For New Development, Product Design is the activity of choosing (and possibly modifying) the (software) architecture for a product, for populating ("instantiating") the architecture with named components and allocating requirements to the architectural components. For large projects, this activity ends with the Preliminary (or Product) Design Review (PDR). (The Product Design is often chosen during the "Define the Approach" activity in ATISD's project management process. For a large project, the Product Design is refined during a "Product Design" activity performed during the "Execute the Plan" activity.) The Formal Acceptance Test strategy is also defined during Product Design. (This strategy may influence the design, e.g., by adding the means to inject and capture data to facilitate testing.) Specific test cases are developed during Detailed Design. Test procedures are usually written during Code and Unit Test. For some projects these activities are performed sequentially.

For Maintenance, the software architecture is already defined. (If major modifications to the architecture are needed, the New Development process is followed.) Product Design just identifies those modules to be modified and/or added to the product, and allocates the new/modified requirements to these modules. Any new modules added comply with the existing architecture. The Formal Acceptance Tests are also planned during Product Design. This includes (1) reviewing existing test cases and selecting ones to test the planned modifications and to verify system integrity (regression tests), and (2) defining new test cases to test the new features which will be added. The actual test procedures are usually written during the Implementation phase.

product development plan (PDP) - a Project Management Plan for a developing a new product. This plan includes information similar to that included in the Product Sustainment Plan (PSP) and the Product Implementation Plan. (See Product Sustainment Plan and Product Implementation Plan.) We recommend, however, that any version-specific information be clearly separated from the process definition in PDPs as well. Equivalent to an SDP on a software project.

product implementation plan (PIP) - a plan which identifies the specific product components to be built during a particular project and the milestones, the schedule and the resources required. A form of a Project Management Plan.

product life cycle - a product typically goes through these phases: concept definition (feasibility is verified), full scale development (the initial version of the system is built and deployed), transition (the responsibility for product upkeep is passed to another organization), sustainment (the product is repaired and enhanced and new versions are released) and retirement (the product is taken out of service, "decommissioned").

product sustainment plan (PSP) - this is a plan defining how a product will be maintained. It defines the process which will be (re)used to produce all versions of the product, but does not describe each version. (At the time the PSP is written, the versions are not defined! The Build Implementation Plan (BIP) defines each version.)

product trouble report (PTR) - another (and more general purpose) name for STRs.

product vision - a short document defining the purpose and features of the product. It is written from a user perspective. The vision statement defines what the product is, and what the product is NOT. It defines the most essential features and functions of the product, without which the product will be considered unacceptable or a failure. It also bounds the scope of the product. Also includes planned upgrades, operational constraints (human safety), constructing portions for later reuse, compatibility with other products, etc. List no more than 10 items, preferably 5 or less.

Vision statements guide the team without excessive detail, channeling their creative energies in the same general direction. This channeling allows the team to innovate and adapt their designs within broad, commonly agreed constraints. The vision statement can also be used to record the developer's understanding of the customer's requirements and to communicate these back to the customer soon, before substantial effort has been expended.

production process - the sequence of engineering activities performed to specify, build and test a product. The project life cycle is a part of this process. The production process is defined by a set of standards and procedures.

production sequence - same as project life cycle

program - a program may contain several projects and it is run strategically.

program plan - establishes the basis for operating the entire program as defined by the contract. A program typically last several years. The Program Plan describes the management approach for a task order contract. The plan describes the type of work to be done, rate structure, organizational structure, and the process for managing task orders. This process covers defining task, negotiating task and contract modifications, assigning responsibilities and resources for task, and monitoring and reporting the status and progress of the task. A key part of the task management process is to analyze each task

requested by the Customer to determine if that task is subject to the full production process and so requires a Project Management Plan (PMP). (For most Task Order Contracts we deliver only services and so the full process does not apply to each task order.) If a tangible product is delivered then the full standard process applies to the task order. The Program Plan is called the Product Sustainment Plan for maintenance contracts. (Also see Build Implementation Plan (BIP)).

project - a collection of people equipment, tools, and facilities assembled for the purpose of achieving specific objectives within specified cost and schedule constraints. Projects are an undertaking requiring concerted effort, they follow an ordered list of events or procedures that are to take place and they are managed tactically. Every project has a defined scope, specific deliverables, and start and completion dates. Project deliverables can be products, services or both. In some cases the services consist of delivering a specific number of labor hours in a special skill category. (See **task order contract** and **program plan**.) For the ASEPH, a project begins when we receive an approved offer from the Customer.

project characterization - our interpretation of the customer's statement of work - engineer's view.

project goals - the project goals defines the success criteria for a project. They record the expectations and business objectives for the project. These are written from a developer viewpoint. They cover what the project must produce and the approximate cost and schedule for the project. These goals guide the team and maintain focus during planning and executing the project. Typical items included in the project goals are: due (completion) date, expected (negotiated) cost, contract type (T&M, CP, FFP), staffing level (in FTE), build/release sequence (if appropriate), acceptance test concept (scope, formality), delivery concept (deployment, distribution), constraints (e.g., hard launch date, "politics", CMM Level), major risks ("technical show-stoppers", regulatory approvals/certifications, indemnification, liquidated damages).

project library - each project will have a location where all project team members can have access to the project documents (ASPEH, Project Notebook, Unit Development Folders, etc.).

project life cycle - same as project life cycle

project life cycle model - a particular sequence of production activities (requirements analysis, product design, implementation and test). Examples are Waterfall, Incremental, Phased, Evolutionary and Spiral. (Maintenance uses the Evolutionary model.) The general sequence for each type of life cycle is defined by the Software Process Architecture. The specific characteristics of each activity (scope, products, cost and schedule) in the sequence are defined when the project is planned. (Planning "instantiates" the elements of the sequence.)

project manager - the person responsible for planning and executing a project.

project notebook – a binder that contains some of the project records. Usually the records that cannot logically be stored in another location.

project plan(s) - documents defining the methods, tools, SEE/STE, facilities and skills to be used to build, modify, sustain, install, implement or transition a product. (Products can be software, data bases, manuals, studies or evaluations.) Project plans are usually prepared by tailoring the standard process, defined in the ASEPH, for a specific project type. For large projects, project plan(s) are usually split into several separate plans (covering development, configuration management, quality assurance, risk management, etc.)

Project plans include Project Management Plans (PMP), Program Plans (PP), Product Sustainment Plan (PSP), Build Implementation Plan (BIP), Transition Plans, Fielding Implementation Plan (FIP), Quality Assurance Plan (QAP), Configuration Management Plan (CMP),etc. The PMP is the Project Plan for New Development. The Transition Plan is the Project Plan for a Transition project. The PSP and PIP (or the SSP and BIP) jointly constitute the Project Plan for a Maintenance project. The term "Project Management Plan" is most often used in lieu of SDP or SSP when the project includes the development or modification of hardware.

project records – the collection of all project data, including the project's defined (tailored) process, procedures and standards; the project's plan(s), and plan status; product documents (specifications, designs, test plans and procedures); and associated artifacts such as meeting minutes, action items, unit development folders, and peer review reports. The project records are used to actively manage and perform the project's tack. The project records may be maintained electronically or in hard copy form. The individual items comprising the project records can be stored anywhere in the project's facilities (because the items must be readily accessible to workers on a daily basis). Each project must maintain n index to the items and artifacts comprising its project records. (See ASEPH Compliance Guide.)

project team - the team that builds the product. Team includes: Project Manager, Engineers, Quality Assurance Specialist, and Configuration Management Specialist.

project training strategy - defines the nature and scope of the training activities needed for the project. It is prepared in ENG1 and used to estimate the resources needed. It is also used in ENG2 to write the actual plans.

prototype - an incomplete, terse/skimpy software product that is always discarded. Prototypes are built to answer questions (quantify performance), prove concepts/ feasibility, etc. They lack many of the features needed for the final product and usually are incompatible with the architecture of the final product. (For example, they may be coded in C++, while the product is coded in Ada.) This means that they are by their nature difficult or impossible to evolve into the final product and so you should not attempt to do so.

prototype build - a Prototype Build is a build that will not be used as a part of a fielded system. A Prototype may be a build developed to learn the potential customer's

reaction. Prototypes may demonstrate one or more facets of system behavior for tests with the intended users. Another use of prototypes is to verify feasibility and performance characteristics of proposed concepts. Prototypes thus help to reduce uncertainties in the requirements and development risks. Prototypes should involve end users who can detect operational problems. Prototypes are not part of a fielded system and so they are not baselined by Configuration Management. Prototypes are built by implementing the ASEPH process activities as tailored by the project.

purchase order - a document sent to a vendor from SAIC requesting that certain items be sent to SAIC.

purchase requisition - a document filled out by a requester and is then routed to receive approval signature from the appropriate personnel. This document may be paper or electronic (SAP).

quality assurance strategy - defines the nature and scope of the QA activities needed for the project. It is prepared in ENG1 and used to estimate the resources needed. It is also used in ENG2 to write the actual plans.

quick look test report - a report detailing what has happen during a short period of time (usually a day) of testing.

receiving report - a report contained in the SAP system which the "Receiver" fills out if a shipment is received and the items shipped are not the items specified on the Purchase Order.

release - a working version of the final product which implements some capability useful to the customer, even though it lacks the full capabilities specified/planned for the product. It is a working version that IS DELIVERED to the user. It will become part of the final product via (1) adding new pieces, (2) extending/evolving the current pieces, or (3) both. The key difference between a release and a build is that the developer must support the release after it is delivered. (e.g., run a help desk, answer questions, etc.) Thus, phased development costs more effort and money than incremental development.

request for proposal (RFP) - a document from a potential customer asking for companies to submit a solution to a problem they want solved.

resource loaded networks (RLN) - is a logical time-phasing of activities necessary to accomplish the entire project scope. Each activity in the network is characterized by scope, logical relationships, duration, and resources. The RLN is the means used to integrate activities and resources into a meaningful arrangement depicting the timing of the critical activities that will satisfy the customer's requirements.

reverse engineering - the process of analyzing software (typically using tools) to determine requirements satisfied by software and its design and structure. This information is used to understand the functionality of software which must be extracted and re-implemented or maintained.

reviews - there are two types of reviews: (1) Technical Review - a one on one review of a document or work product by a person who is technically qualified to review the work. (2) Peer Review - a group of persons reviews the work product and gets together in a meeting to discuss the defects found.

revised ASEPH - a new version of the standard process document.

rework request - a written request from Upper Management, Contracts or the Customer to the Project Manager requesting that the plans and/or the estimates be redone. The request should state who is making the request, the date of the request and the reason for the request.

rigor - determines how much documentation and vigilance should be taken on a project. Higher rigor projects typically produce more paperwork, have more reviews and are subject to greater scrutiny.

rough estimates - estimates prepared by the Project Manager early in the life of the project. They are made based on the information the Project Manager has available such as the SOW, Project Characterization Form, Project Rigor, etc. These estimates are will be given to the engineers to aid them in preparing more detailed estimates.

schedule - a document showing the task for a project and when the task will be started and completed.

screening - the activity of analyzing and validating "raw" trouble reports to identify the validated, unique trouble reports. Screening produces the information contained in ECAs. (The actual ECA is not always written at the time the screening is done. It should be however.)

SEI (Software Engineering Institute) - is a federally funded research and development center sponsored by the Department of Defense through the Advanced Research Projects Agency (ARPA). The SEI contract was competitively awarded to Carnegie Mellon University in December 1984. It is staffed by approximately 250 technical and support people from industry, academia, and government.

The SEI mission is to provide leadership in advancing the state of the practice of software engineering to improve the quality of systems that depend on software.

The SEI expects to accomplish this mission by promoting the evolution of software engineering from an ad hoc, labor-intensive activity to a discipline that is well managed and supported by technology.

services - is delivered by expending labor and is dynamic is nature. Examples are: delivery of training courses, installation of equipment, etc.

shipping - done in production facility (plant). Follows production of the products and the Factory Acceptance Tests. We simply take orders (or use the list of deliverables we negotiated), package items and ship them. We never leave our production facility. The customer is responsible for installing product.

shipping request form - a standard form available in shipping and receiving that is used to request that an item or items be shipped to another location or a Customer site.

signed receipts - a receipt signed by the Customer showing that they accept the product.

site acceptance test (SAT) - a (usually formal) test conducted after a product has been installed at a user site for the purpose of verifying the correct operation of the installed product. Also see Field Test.

size - the amount of product produced or consumed. Usually measured in lines of code, functions points, records, tokens, etc.

soft deliverables - products and services whose requirements are defined by the developer. Examples are trade studies, technical reports, and even computer programs. The product is not subject to stringent constraints and customer acceptance is based only on general, high level criteria.

software architecture - for the purposes of the ASEPH, software architecture means the set of hardware (platforms) and software components, their partitioning (structures arrangement and relations), and the rules governing the interactions (data transfer, control, error handling) between these components, and between these components and external entities (users and other systems). It is a framework defining common elements and constraints of a (software) product. An architecture consists of documentation which:

(1) identifies all hardware and software components of the system
(2) describes the (static) structure and relations between the components, including
    (a) which software components reside on which hardware components, and
    (b) the nature and scope of each component or subsystem (i.e., the broad functionality of the components, e.g., all data management functions could be allocated to a particular component).
(3) defines the rules for the dynamic interaction of the components, including rules for error detection, propagation, handling, for inter-process communication and synchronization, human/machine interactions ("look and feel"), internal coordinate systems and physical units.
(4) defines design and construction constraints. For example, a design constraint is that every module has a prolog with some standard format. A possible construction constraint is that all code is written in Ada. (Specification of internal coordinate systems and physical units are sometimes considered design constraints. We prefer to consider them as part of the inter-module interfaces.)

software development plan (SDP) - a Project Management Plan for a New Development project which produces software. Sometimes the SDP is called the Project Plan, especially when hardware development is included as part of the project. A more general name is Product Development Plan (PDP).

software engineering environment (SEE) - the collection of computer equipment, tools and documentation used to modify, compile, link and test software for a system. Also see software test environment.

software integration and test (SWIT) - the combining of unit-tested software components into larger subsystems (and eventually into the entire system) for the purpose of verifying correct operation. The testing performed during SWIT emphasizes system-level functions and interfaces between components. SWIT is usually performed by SWIT is done by an independent test team, i.e., by people who did not develop the software being tested.

software process architecture - a generic description of a production process. It consists of a set of activities and the artifacts they produce and consume, plus rules and guidelines for customizing (tailoring) these to the needs of a specific product line and project. The description specifies the relations (connectivity and time sequencing) of the activities. Connectivity is accomplished via artifacts which flow between activities. To produce a particular system, the Project Manager defines which sequence of activities should be used and documents the sequence in the project's plan (SDP or equivalent). Also see process flow and tailoring.

software process improvement - a process is the logical organization of people, automated support, procedures, and standards into work activities designed to produce a specified end result. A project's process embodies methods, tools, policies and procedures.

Software Process Improvement (SPI) is the organized activity of defining, infusing, and improving the processes used by individual projects and organizations to develop software. SPI is a metaprocess since it monitors and alters entire classes of project processes. The Software Engineering Institute (SEI) embraces the concept of statistical process control. When a process is under control, repeating the work in the same way will give roughly the same results. To obtain consistently better results, it is thus necessary to improve the process. If the process is not under statistical control, however, sustained progress is not possible. Thus, if we improve the process, product quality will improve too.

software process improvement plan (SPIP) – a plan written annually to document the process improvement activities for the coming year It contains the training plan, and schedule and budget for the whole program.

software support plan (SSP) - a Product Sustainment Plan for software. It defines how a product line will be maintained. The SSP defines the process that will be used for all builds of a particular system, but not the content of a specific build. (This distinction has not always been made at SED, resulting in overly large SDPs prepared for every build. The use of the terms SSP (PSP) and BIP (PIP) separates build-independent and build-specific information into two plans, eliminating the needless repetition of identical information. The process defined in the PSP for a system is "reused" for all builds of the product.)

software test environment (STE) - the collection of computer equipment, tools and documentation used to integrate and test software for a system. For some systems, the STE and the software engineering environment (SEE) are identical. For other systems, the STE has additional equipment, tools, etc. to perform stress testing, verify interfaces to other systems (interoperability), etc.

software trouble report (STR) - a report submitted by an end user describing a problem with a fielded product. Also called Product Trouble Report (PTRs). Sometimes the STR really requests a new capability and so is like a BCR.

staffing profile - a document showing how many people of each labor type will be required during each phase or timeline for the project. Ideally the Project Manager attempts to keep the personnel count as level as possible for the life of the project with very few peaks and valleys.

standard - a documented description of an artifact, giving its purpose, intended audience, content and (optionally) a format.

standard process - a defined and documented business and engineering process used by all projects in an organization.

standard product - a product designed for sale to multiple customers at a standard price. These can range from "shrinkwrapped" products installed by the buyer up to complex, products configured and installed by the developer. An example of the latter type is a plant monitoring system.

statement of work (SOW) - CMM definition is that this is the customer's perspective or perception of the product. This is usually formalized using one or more documents: Statement OF Work (SOW), Contract Data Requirement List (CDRL), Contract Line Item Number (CLIN). For smaller projects a simple document suffices which is a task to be performed or a list of deliverables.

status briefing - a meeting held at least quarterly where the Group Manager is briefed on the status of the projects.

student answer sheet - a sheet filled out by the student recording their answers to a test given at the beginning or the completion of a course.

student training record – a report that can be generated by the training database. The report shows all the courses required for the employee based on the specified roles. It also shows the courses taken, the required courses yet to be taken and any supplemental training the employee has.

study plan - a plan to perform a study or analysis. This plan is typically very short and identifies the purpose, deliverables, schedule (milestones and dates), estimated resources and Project Manager and key staff (if any).

support task - a task usually done under a Task Order contract which merely delivers labor hours. The task has no deliverables and the staff performs work activities as directed by the Customer and uses the Customer's process.

swit test procedures - the test procedures used by the engineers to wring-out all the bugs from the integrated system before it enters Dry Run testing. These procedures are usually not as formal as the test procedures used for FAT and test much more then the FAT test should test (although the FAT test can be run as part of SWIT testing).

tailoring - refers to the selection and customization of those activities and artifacts of the standard organizational process needed to accomplish a particular project. Tailoring is performed by the Project Manager, assisted by the project staff, during the planning process. (Tailoring is usually done during the "Define the Technical Approach" activity, although it may be done elsewhere depending on the project type.) The Project Manager considers factors such as the characteristics of the particular weapon system, the type of work to be done and the desires of the Customer.

Tailoring is not unconstrained. Every project performed at the ATISD must comply with ATISD policies and so must meet certain minimum essential criteria. For example, every project must have a documented plan which is approved by the project participants. These constraints are embodied in the standard organizational process (specifically the Project Management process flow) described in the ASEPH.

task assignments - maps person to work package

task order contract - a contract where a basic contractual arrangement is established setting a general scope, rate structure, and reporting requirements. Work is assigned and negotiated as separate task which are described by "task orders". Multiple task orders may be included under a single contract modification.

technical data package - the collection of engineering documents, drawings, diagrams, etc. which describe a product. This information provides a description of the product that is complete enough to allow additional copies of the product to be built and tested. This information also provides the basis for producing a modified version of the product. Typical contents of the TDP include: requirements, product architecture and design, detailed design, and software development folders.

test report - a report written by Quality Assurance during the Acceptance testing of a product. The test shows the test run and the results of the test.

test strategy - the documentation of the testing "plan" for a project. The strategy is written early in the life of a project for initial planning and estimation efforts and should be agreed to by the Customer. It is further defined in more detail in the Testing Plan in the Detailed Planning Phase of the project.

It is a brief description of types and scope of product testing to be performed. This covers the types of tests (e.g. functional, interface, performance, stress, endurance [reliability], regression), the test configuration (product software, equipment, test tools, dummy data), risks, and organizational responsibilities. The test strategy must cover

product acceptance testing, and may also cover integration testing. For example, if special equipment is needed, certain tests may be deferred from Factory Acceptance Test to Site Acceptance Test. Alternately equipment emulators may be used during integration and FAT, and the actual equipment used during SAT. Integration test strategy is usually closely tied to the build sequence (specified by the Project Team) and the delivery sequence (specified by the Customer).

third party - an organization outside the control of the developer or the customer, such as OSHA, SEI, ISO, FDA, etc. Typically these organizations have no financial stake in the project or the product and so are not motivated to act in an expedient or efficient manner.

top level design (TLD) [noun] - the products of the Top Level Design activity. A better name for these products is the Product Specification. This specification describes the product, which meets the requirements.

top level design (TLD) [verb, gerund] - the activity of identifying and naming specific instances of the architectural components, and allocating specific requirements to each of these components. Top Level Design "instantiates" the architecture for a specific system or product. For example, in a client/server architecture, a specific communications protocol, say Open System Interconnect (OSI), may be chosen for the Communications Manager module and so the module would be named "OSI_Comm_Manager". Also see Top Level Design (TLD) [noun].

training plan - a plan identifying the knowledge and skills needed by a project team to perform some specified type of work. This information includes the courses required, the number of staff members planned to take each course, and the resources required (student labor and expenses). [Instructors are usually funded via the organization's Annual Training Plan. If special courses are purchased from outside vendors, however, these costs should be included in the project's training plan.] For most projects, training needs are modest and so this information can be recorded in a section of the Project Plan (PDP, PSP, SDP or BIP).

transition - the process of transferring responsibility for the Maintenance of a product from the original developer to a sustaining organization. This occurs in three stages: preparation, installation and activation. During Preparation, the organization analyzes the product and the organization's existing capabilities to determine the methods and resources needed to perform Maintenance. A Transition Plan is written to acquire new capabilities and/or to augment the existing capabilities sustaining organization. During Installation, the Transition Plan is executed. The necessary capabilities are installed (SEE, STE), the staff is trained, and SDP is written to define the methods and use of the SEE/STE, and the product is placed under configuration control in the Maintainer's facility. Optionally, a "demonstration build" is produced by the sustaining organization to verify its ability to correctly modify the product. During Activation, the owner reviews the results of the Installation phase, and decided whether or not to give responsibility for Maintenance to the organization.

The Transition process is planned and managed as a project using a Scope of Work (for the Preparation phase) and a Transition Plan (for the Installation phase). Also see Transition Plan.

transition plan - a plan defining how the capability to sustain a product line will be established. This plan identifies tasks to acquire tools and equipment, expand facilities, train staff, and define the Maintenance process which will be used to produce new versions of the product. The Transition Plan is written during the preparation phase and is executed during the Installation Phase.

upper management (UM) - the persons responsible for supervising the activities of Project Managers and for maintaining the SAIC infrastructure.

unit - a software unit performs a single well-defined function, can be developed by one person, and is typically 100 to 300 source instructions in size.

validation - (see IEEE STD 610)

version description document - the build directive used to create the version of the product shipped to the customer.

vision - see product vision

verification - (see IEEE STD 610)

waterfall - classical development method. Follows the sequence analyze, design, code and test. In true waterfall there is not iterations.